AMENDMENTS TO THE CLAIMS:

This listing of claims replaces all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1. (Currently Amended) A method comprising:

receiving successive frames carrying data in timeslots, the timeslots being assigned to channels so that data for the channels comprises interleaved data;

aggregating the data from successive frames for each of a predetermined number of the timeslots;

mapping the aggregated data, by timeslot, to produce a timeslot-based map, the mapping comprising storing the aggregated data in a first buffer; and

remapping the aggregated data in the timeslot-based map to produce a channel-based map by storing the aggregated data stored in the first buffer in a second buffer according to timeslot-remap information to re-order the aggregated data stored in the timeslot-based map, the timeslot-remap information comprising a table, and the table comprising an address and an increment value associated with each of the timeslots in a frame.

2. (Original) The method of claim 1 wherein, in the channel-based map, the data for the channels are grouped together, by channel, in the order in which the data were received.

Claims 3 to 5 (Cancelled)

6. (Currently Amended) The method of claim [[5]] 1 wherein remapping comprises:

remapping the aggregated data for each timeslot, in sequential order;

reading bytes of aggregated data for the timeslot from the first buffer;

determining a destination address in the second buffer according to the associated address in the table; and

storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address.

7. (Currently Amended) The method of claim 6 wherein remapping ~~the aggregated data for each timeslot~~ comprises:

~~reading bytes of aggregated data for the timeslot from the first buffer;~~

~~determining a destination address in the second buffer according to the associated address in the table;~~

~~storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address;~~

incrementing the destination address by the associated increment value;

storing a next byte of the aggregated data for the timeslot in the second buffer at the incremented destination address; and

repeating incrementing and storing a next byte until all of the bytes of the aggregated data for the timeslot have been stored in the second buffer.

8. (Currently Amended) The method of claim 1 [[7]], further comprising:

maintaining two copies of the table, the two copies including an active table and a shadow table.

9. (Original) The method of claim 8 wherein the table used during the remapping the aggregated data for each timeslot is the active table.

10. (Original) The method of claim 9, further comprising:

enabling a re-configuration of the table by a modification of the shadow table; and

providing an indicator that the table has been re-configured.

11. (Original) The method of claim 10 wherein remapping further comprises:

prior to reading any bytes of data from the first buffer, determining from the indicator if the table has been reconfigured; and

if it is determined that the table has been reconfigured, swapping the shadow table with the active table.

12. (Original) The method of claim 10 wherein the indicator comprises a signal to indicate that a reconfiguration of the table has occurred.

13. (Original) The method of claim 1 wherein the successive frames are T1 frames.

14. (Original) The method of claim 1 wherein the successive frames are E1 frames.

15. (Original) The method of claim 1 wherein the successive frames are J1, frames.

16. (Original) The method of claim 1 wherein the data includes High-Level Data Link Control data.

17. (Original) The method of claim 1 wherein the data includes Asynchronous Transfer Mode data.

18. (Original) The method of claim 1 wherein the data comprises interleaved data and non-interleaved data.

Claim 19 (Cancelled)

20. (Currently Amended) An article comprising:

a storage medium having stored thereon instructions that when executed by a machine result in the following:

for successive frames carrying data in timeslots assigned to channels so that data for the channels comprises interleaved data, aggregating the data from successive frames for each of a predetermined number of the timeslots;

mapping the aggregated data, by timeslot, to produce a timeslot-based map, the mapping comprising storing the aggregated data in a first buffer; and

remapping the aggregated data in the timeslot-based map to produce a channel-based map by storing the aggregated data stored in the first buffer in a second buffer according to timeslot-remap information to re-order the aggregated data stored in the timeslot-based map, the timeslot-remap information comprising a table, and the table comprising an address and an increment value associated with each of the timeslots in a frame, remapping comprising:

reading bytes of aggregated data for the timeslot from the first buffer;

determining a destination address in the second buffer according to the associated address in the table; and

storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address.

21. (Original) The article of claim 20 wherein, in the channel-based map, the data for the channels are grouped together, by channel, in the order in which the data were received.

22. (Currently Amended) A ~~network~~ processor comprising:

a serial interface;

a network processor engine of multiple execution threads coupled to the serial interface;

wherein the serial interface is configured to process successive frames carrying data in timeslots, the timeslots assigned to channels so that data for the channels comprises interleaved data, the processing including aggregating the data from successive frames for each of a predetermined number of the timeslots and mapping the aggregated data, by timeslot, to produce a timeslot-based map; and

wherein at least one of the multiple execution threads, during execution, operates to remap the aggregated data in the timeslot-based map to produce a channel-based map by storing aggregated data stored in a first buffer in a second buffer according to timeslot-remap information by using the timeslot-remap information to re-order the aggregated data stored in the timeslot-based map, the timeslot-remap information comprising a table, and the table comprising an address and an increment value associated with each of the timeslots in a frame; wherein to remap comprises:

reading bytes of aggregated data for the timeslot from the first buffer;

determining a destination address in the second buffer according to the associated address in the table; and

storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address.

23. (Currently Amended) The ~~network~~ processor of claim 22 wherein, in the channel-based map, the data for the channels are grouped together, by channel, in the order in which the data were received.

24. (Currently Amended) The ~~network~~ processor of claim 22 wherein the successive frames are T1 frames.

25. (Original) The ~~network~~ processor of claim 22 wherein the successive frames are E1 frames.

26. (Currently Amended) The ~~network~~ processor of claim 22 wherein the successive frames are J1 frames.

27. (Currently Amended) The ~~network~~ processor of claim 22 wherein the data includes High-Level Data Link Control data.

28. (Currently Amended) The ~~network~~ processor of claim 22 wherein the data includes Asynchronous Transfer Mode data.

29. (Currently Amended) The ~~network~~ processor of claim 22 wherein the data comprises interleaved data and non-interleaved data.

30. (Currently Amended) A system comprising:

a framer adapted to be coupled to a network;

a network access device coupled to the framer, the network access device comprising a network processor; and

the ~~network~~ processor comprising:

a serial interface;

a network processor engine of multiple execution threads coupled to the serial interface;

wherein the serial interface is configured to process successive frames carrying data in timeslots, the timeslots assigned to channels so that data for the channels comprises interleaved data, the processing including aggregating the data from successive frames for each of a predetermined number of the timeslots and mapping the aggregated data, by timeslot, to produce a timeslot-based map; and

wherein at least one of the multiple execution threads, during execution, operates to remap the aggregated data in the timeslot-based map to produce a channel-based map by storing aggregated data stored in a first buffer in a second buffer according to timeslot-remap information by using the timeslot-remap information to re-order the aggregated data stored in the timeslot-based map, the timeslot-remap information comprising a table, and the table comprising

an address and an increment value associated with each of the timeslots in a frame; wherein to remap comprises:

reading bytes of aggregated data for the timeslot from the first buffer;

determining a destination address in the second buffer according to the associated address in the table; and

storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address;

incrementing the destination address by the associated increment value; and

storing a next byte of the aggregated data for the timeslot in the second buffer at the incremented destination address.

31. (Original) The system of claim 30 wherein, in the channel-based map, the data for the channels are grouped together, by channel, in the order in which the data were received.

32. (New) The article of claim 20 wherein remapping comprises:

incrementing the destination address by the associated increment value; and

storing a next byte of the aggregated data for the timeslot in the second buffer at the incremented destination address.

33. (New) The method of claim 32 wherein remapping comprises repeating incrementing and storing a next byte until all of the bytes of the aggregated data for the timeslot have been stored in the second buffer.

34. (New) The article of claim 22 wherein to remap comprises:

remapping the aggregated data for each timeslot, in sequential order;

incrementing the destination address by the associated increment value;

storing a next byte of the aggregated data for the timeslot in the second buffer at the incremented destination address; and

repeating the incrementing and the storing a next byte until all of the bytes of the aggregated data for the timeslot have been stored in the second buffer.

35. (New) A method comprising:

mapping aggregated data, by timeslot, to produce a timeslot-based map, the mapping comprising storing the aggregated data in a first buffer; and

remapping the aggregated data in the timeslot-based map to produce a channel-based map by storing the aggregated data stored in the first buffer in a second buffer according to timeslot-remap information to re-order the aggregated data stored in the timeslot-based map, the timeslot-remap information comprising a table, and the table comprising an address and an increment value associated with each of the timeslots in a frame, remapping comprising:

reading bytes of aggregated data for the timeslot from the first buffer;

determining a destination address in the second buffer according to the associated address in the table;

storing the first byte of the aggregated data for the timeslot in the second buffer at the destination address;

incrementing the destination address by the associated increment value;

storing a next byte of the aggregated data for the timeslot in the second buffer at the incremented destination address; and

repeating the incrementing and the storing a next byte until all of the bytes of the aggregated data for the timeslot have been stored in the second buffer.